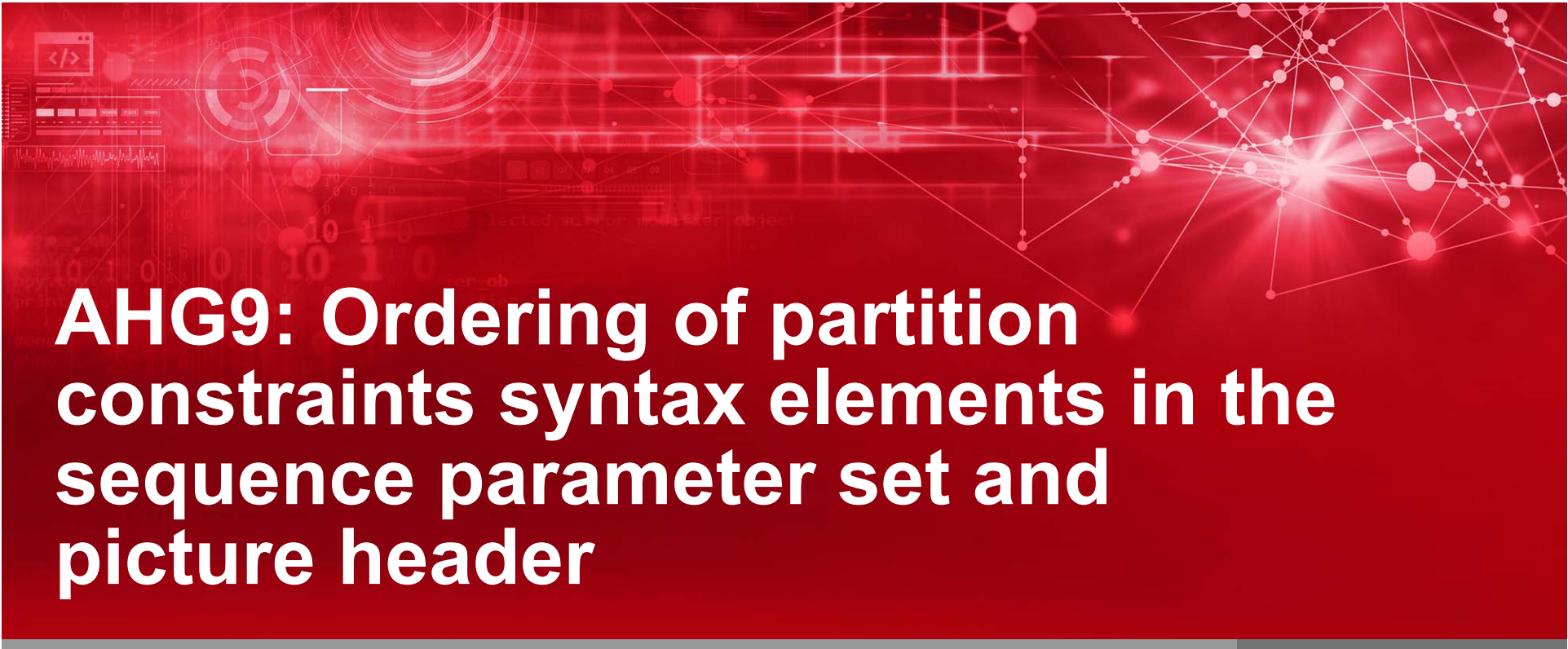


The background is a solid red color. Overlaid on this are various faint, semi-transparent white and light red graphics. These include a network diagram with nodes and connecting lines on the right side, a circular target-like graphic on the left, and some abstract geometric shapes and lines scattered throughout. The overall aesthetic is technical and modern.

AHG9: Ordering of partition constraints syntax elements in the sequence parameter set and picture header

JVET-Q0481

W. Wan, B. Heng

Brussels, Belgium, January 2020 meeting



Introduction

- When there are similar syntax elements for intra and inter coded pictures, the typical convention has been to define the intra coded parameters before inter coded parameters
- The mtt_hierarchy_depth syntax elements are defined in the reversed direction with inter coded parameters defined before intra coded parameters
- Propose to rearrange to avoid potential confusion and interop problems as well as design consistency

Current draft text (SPS syntax)

- Confusing ordering (Intra → Inter → Intra → Inter)

	Descriptor
seq_parameter_set_rbsp() {	
...	
sps_log2_diff_min_qt_min_cb intra_slice_luma	ue(v)
sps_log2_diff_min_qt_min_cb inter_slice	ue(v)
sps_max_mtt_hierarchy_depth inter_slice	ue(v)
sps_max_mtt_hierarchy_depth intra_slice_luma	ue(v)
if(sps_max_mtt_hierarchy_depth intra_slice_luma != 0) {	
sps_log2_diff_max_bt_min_qt_intra_slice_luma	ue(v)
sps_log2_diff_max_tt_min_qt_intra_slice_luma	ue(v)
}	
if(sps_max_mtt_hierarchy_depth inter_slice != 0) {	
sps_log2_diff_max_bt_min_qt_inter_slice	ue(v)
sps_log2_diff_max_tt_min_qt_inter_slice	ue(v)
}	
...	
}	

Current draft text (PH syntax)

- Confusing ordering (Intra → Inter → Intra → Inter)

	Descriptor
picture_header_rbsp() {	
...	
pic_log2_diff_min_qt_min_cb intra slice_luma	ue(v)
pic_log2_diff_min_qt_min_cb inter slice	ue(v)
pic_max_mtt_hierarchy_depth inter slice	ue(v)
pic_max_mtt_hierarchy_depth intra slice_luma	ue(v)
if(pic_max_mtt_hierarchy_depth intra slice_luma != 0) {	
pic_log2_diff_max_bt_min_qt intra slice_luma	ue(v)
pic_log2_diff_max_tt_min_qt intra slice_luma	ue(v)
}	
if(pic_max_mtt_hierarchy_depth inter slice != 0) {	
pic_log2_diff_max_bt_min_qt inter slice	ue(v)
pic_log2_diff_max_tt_min_qt inter slice	ue(v)
}	
...	
}	

SPS syntax changes (no semantic changes)

seq_parameter_set_rbsp() {	Descriptor
...	
sps_log2_diff_min_qt_min_cb_intra_slice_luma	ue(v)
sps_log2_diff_min_qt_min_cb_inter_slice	ue(v)
sps_max_mtt_hierarchy_depth_inter_slice	ue(v)
sps_max_mtt_hierarchy_depth_intra_slice_luma	ue(v)
if(sps_max_mtt_hierarchy_depth_intra_slice_luma != 0) {	
sps_log2_diff_max_bt_min_qt_intra_slice_luma	ue(v)
sps_log2_diff_max_tt_min_qt_intra_slice_luma	ue(v)
}	
sps_log2_diff_min_qt_min_cb_inter_slice	ue(v)
sps_max_mtt_hierarchy_depth_inter_slice	ue(v)
if(sps_max_mtt_hierarchy_depth_inter_slice != 0) {	
sps_log2_diff_max_bt_min_qt_inter_slice	ue(v)
sps_log2_diff_max_tt_min_qt_inter_slice	ue(v)
}	
...	
}	

PH syntax changes (no semantic changes)

picture_header_rbsp() {	Descriptor
...	
pic_log2_diff_min_qt_min_cb_intra_slice_luma	ue(v)
pic_log2_diff_min_qt_min_cb_inter_slice	ue(v)
pic_max_mtt_hierarchy_depth_inter_slice	ue(v)
pic_max_mtt_hierarchy_depth_intra_slice_luma	ue(v)
if(pic_max_mtt_hierarchy_depth_intra_slice_luma != 0) {	
pic_log2_diff_max_bt_min_qt_intra_slice_luma	ue(v)
pic_log2_diff_max_tt_min_qt_intra_slice_luma	ue(v)
}	
pic_log2_diff_min_qt_min_cb_inter_slice	ue(v)
pic_max_mtt_hierarchy_depth_inter_slice	ue(v)
if(pic_max_mtt_hierarchy_depth_inter_slice != 0) {	
pic_log2_diff_max_bt_min_qt_inter_slice	ue(v)
pic_log2_diff_max_tt_min_qt_inter_slice	ue(v)
}	
...	
}	



BROADCOM[®]

connecting everything[®]